

一种改进的混沌粒子群算法

郭明山¹, 刘秉瀚²

(1. 中国移动福建有限公司漳州分公司网络部 福建 漳州 363000 2. 福州大学计算机系 福建 福州 350002)

【摘要】 粒子群算法算法简洁,易于实现,但是传统的粒子群算法易陷入局部极小点。为此,本文在传统的粒子群算法中引入混沌思想用于改善算法性能。通过数值模拟表明,本文算法有比较好的性能。利用本文算法来求解无线网络优化中的加站问题,取得了比较好的效果。

【关键词】 混沌;最优化;粒子群算法

1. 引言

粒子群优化算法是一种新兴的有潜力的演化算法,PSO的优势在于算法的简洁性,易于实现,没有很多参数需要调整,而且不需要梯度信息。PSO是非线性连续优化问题,组合优化问题和混合整数非线性优化问题的有效优化工具^[1]。目前已经广泛地应用于函数优化,神经网络训练,模糊系统指控以及其他遗传算法的应用领域^[2]。

2. 粒子群算法及其研究现状

PSO算法中,粒子的位置代表被优化问题在搜索空间中的潜在解。所有的粒子都有一个由被优化的函数决定的适应值(fitness value),每个粒子还有一个速度决定它飞翔的方向和距离。粒子们追随当前的最优粒子在解空间中搜索。PSO初始化为一群随机粒子(随机解)。然后通过迭代找到最优解。在每一次迭代中,粒子通过跟踪两个“极值”来更新自己。一个是粒子本身所找到的最优解,称为个体极值;另一个极值是整个种群目前找到的最优解,称为全局极值。在每一次迭代中,粒子通过跟踪个体极值和全局极值来更新自己。PSO算法的数学过程表示如下:

假定搜索的空间为D维,令NUM表示粒子总数,第i个粒子位置向量表示为 $X_i=(x_{i1}, x_{i2}, \dots, x_{iD})$,速度向量表示为 $V_i=(v_{i1}, v_{i2}, \dots, v_{iD})$ 。粒子经历过的最好位置记为 $p_i=(p_{i1}, p_{i2}, \dots, p_{iD})$,也称为 p_{best} 。在群体中所有粒子经历过的最好位置用 $p_g=(p_{g1}, p_{g2}, \dots, p_{gD})$ 表示。

粒子在每一次迭代找到上述两个极值后,对自己的速度和位置进行如下更新:

$$V_{id} = wV_{id} + c_1 \text{rand}_1(p_{id} - x_{id}) + c_2 \text{rand}_2(p_{gd} - x_{id}) \quad (1)$$

$$X_{id} = X_{id} + V_{id} \quad (2)$$

其中: $1 \leq i \leq \text{NUM}$, $1 \leq d \leq D$; c_1, c_2 称为加速因子,通常取 $c_1=c_2=2$ ^[3]; rand_1 和 rand_2 为两个在 $[0, 1]$ 范围内变化的随机函数; w 为速度惯性因子,研究表明^[15]较大的 w 有利于跳出局部极小点,较小的 w 有利于算法收敛。随着迭代进行, w 由最大 w_{\max} 线性减小到最小 w_{\min} ,即:

$$w = w_{\max} - n \frac{w_{\max} - w_{\min}}{N} \quad (3)$$

其中: n 为当前迭代次数, N 为总迭代次数。速度惯性因子 w 通常是从0.9到0.1线性地减少,即 $w_{\max}=0.9, w_{\min}=0.1$ 。

PSO算法的主要缺点是易陷入局部极小点。为克服该缺点,不少学者提出了改进方法。一些改进的PSO算法存在着较复杂,不易使用的缺点;而另一些改进的PSO算法同样存在着计算复杂度高,收敛慢的特点。

基于惯性权值的改进,Shi Y^[4]等提出了使用模糊控制系统自适应调整惯性权值和针对动态优化问题的随机惯性权值等方法,这些改进方法虽然提高了收敛速度同时在单峰问题上取得更高的性能,但在解决多峰值函数问题时,容易陷入局部最优并且实现也比较困难。

基于加速因子的改进,Ratnaweera A^[5]等对加速因子做了一些较为深入的研究并提出了相应的改进算法。实验结果表明了改进算法确实加速了算法的收敛,在单峰问题的测试表现优异,

却付出了算法容易陷入局部最优的代价,而且在多峰值问题的测试中容易过早收敛。

王岁花^[6]等提出了一类新颖的PSO算法,该算法在基本PSO算法的粒子位置更新公式中增加了一个积分控制项,积分控制项根据每个粒子的适应值决定粒子位置的变化,改善了PSO算法摆脱局部极小点的能力^[6]。

柯晶^[7]等提出了一种改进粒子群优化算法(MPSO),MPSO同时采用局部模式压缩因子方法和全局模式惯性权重方法以获得相对较高的性能,针对PSO算法可能出现的停滞现象,MPSO引入了基于全局信息反馈的重新初始化机制^[8]。

杨俊杰等^[9]把混沌优化搜索技术引入到PSO算法中,提出了基于混沌搜索的粒子群优化算法。

此外,Lovbjerg M等^[10]提出杂交(hybrid)PSO算法,Ciuprina G等^[11]提出智能PSO算法, Van den Bergh F等^[12]提出协同PSO算法等。

3. 改进的混沌粒子群算法

为此,本文提出了改进的混沌粒子群算法,在粒子群算法中引入混沌思想从而改善算法性能。实验结果表明,本文算法效果较为理想。

(一) 搜索空间中待调整变量的确定

假设搜索空间为D维,在搜索过程中,为达到比较好的搜索效果,每次只调整其中几个变量,每次调整的变量由以下方法确定。

定义布尔变量 $\rho=(a_1, a_2, \dots, a_D)$,其中 $a_i \in \{1, 0\}$, $1 \leq i \leq D$ 分别代表对应的第i维的变量是否需要改变,然后作如下运算^[13]:

$$S = a_1 2^{-1} + a_2 2^{-2} + \dots + a_D 2^{-D}, a_i \in \{0, 1\}, 1 \leq i \leq D \quad (4)$$

对于式(4),即使 a_1, a_2, \dots, a_D 全都为1,而搜索空间维数趋向无穷,根据无穷等比级数性质,有:

$$\lim_{D \rightarrow \infty} S = 1 \quad (5)$$

因此,对于一个搜索空间有限的寻优问题,式(5)必有 $0 < S < 1$ 。根据这种思想,当每次 S 在 $(0, 1)$ 区间内任取一随机数,都有一个 n 维的布尔变量与之对应,因此可根据 S 来确定该次搜索空间中需要调整的变量。

S 可以按照Logistic映射进行更新。如下:

$$S = 4.0 \times S \times (1.0 - S) \quad (6)$$

(二) 混沌粒子群算法

PSO算法在寻优过程中不同程度地存在早熟问题而易陷入局部最优解,究其原因,在粒子群体的每一次迭代寻优过程中,将至少有1个粒子处于不动状态,假设该粒子为 i ,则有,群体中其他粒子逐渐靠近该粒子,当存在一个粒子,其位置距离该不动粒子足够近时,相当于 $c_1=c_2=0$,则粒子将一直以先前的寻优方向减速飞行,因此只能搜索有限的区域,很难找到问题最优解,从而在很大程度上减弱了此粒子的寻优功能^[14]。因此可以利用混沌的遍历性,对距离足够近而引起搜索能力下降的粒子赋予混沌状态搜索从而跳出局部最优解。

本文提出的混沌粒子群算法思想如下:首先混沌初始化粒

子群中粒子的位置,保证在搜索空间的大范围内初始化粒子群中粒子的位置;然后调用基本的 PSO 方法,当粒子群中出现有多个粒子距离足够近的情况,则对该粒子赋予混沌状态搜索。

本文中为简化算法,混沌迭代方程选用 Logistic 方程。

假设寻优问题的目标函数为: $\min f(x_i) (i=1,2,\dots,D; x_i \in [a_i, b_i])$, 则混沌粒子群算法基本步骤如下:

步骤 1: 参数初始化: 学习因子 c_1 和 c_2 , 惯性因子 w , 粒子群体规模 NUM, 最大进化代数 N_{\max} 和 w_{\min} , 以及最大速度限制 w_{\max} , 初始化混沌变量 S 。注意混沌变量不包括混沌迭代方程的 5 个不动点 $(0, 0.25, 0.50, 0.75, 1)$ 。

步骤 2: 令 $t=0$, 随机生成 D 个不同轨迹的混沌变量 $\text{chaos}_i(t), 1 \leq i \leq D$, 其中 D 为搜索空间维数。 t 表示第 t 次混沌搜索。

步骤 3: 利用混沌变量 $\text{chaos}_i(t), 1 \leq i \leq D$ 对粒子位置混沌赋值。注意必须将混沌变量区间 $[0, 1]$ 映射到对应变量的取值区间。

步骤 4: 计算粒子的适应值。将粒子 i 目前的位置设置为 P_i 。如果当前粒子 i 的适应值为当前最优, 则设置 $p_i = P_i$ 。

步骤 5: 判断粒子群算法是否满足收敛准则, 若满足, 则执行步骤 10, 否则顺序执行下一步。

步骤 6: 根据基本的 PSO 算法对粒子的速率和位置进行更新。首先计算各粒子的飞行速度, 如果飞行速度小于给定的最大速度, 则更新粒子的当前位置; 否则取最大速度, 然后更新粒子的当前位置。

步骤 7: 如果当前粒子群中出现有多个粒子距离足够近的情况(重合), 则一个粒子不变, 其他粒子赋予混沌运动。混沌运动按如下方式进行:

首先由 S 确定该粒子中需要调整变量。

如果需要调整某粒子第 k 维的变量, 则利用混沌变量 $\text{chaos}_k(t) (1 \leq k \leq D)$ 对粒子位置混沌赋值。注意必须将混沌变量区间 $[0, 1]$ 映射到对应变量的取值区间。

步骤 8: 如果新粒子 i 的速度为个体历史最优, 则设置为 P_i ; 如果为粒子群历史最优, 则设置为 P_g 。

步骤 9: 返回步骤 5 执行。

步骤 10: 输出运行结果, 结束程序。

4. 数值模拟

本文将以 2 个基准测试函数的最优值为例, 通过仿真计算来比较评价混沌粒子群算法的性能。

(一) Rosenbrock 函数

一般形式的 Rosenbrock 函数^[13]为:

$$\min f(x) = \sum_{i=1}^{n-1} [100(x_{i+1} - x_i)^2 + (x_i - 1)^2] \quad x_i \in [-100, 100] \quad (7)$$

Rosenbrock 函数为经典测试函数, 其全局极小点位于一个狭长的、抛物线形状的极为平坦的函数曲面山谷中, 一般来说, 找到 Rosenbrock 函数全局极小点所在的山谷并不难, 但要收敛到全局极小点则十分困难, 尤其在低维情况下^[13]。

实验中, 该基准函数的自变量维数取 10。粒子群算法中, 取粒子数为 50, $c_1=c_2=20$, $w_{\max}=0.9$, $w_{\min}=0.1$, 迭代次数取 1000 次。遗传算法中, 取迭代次数为 1000 次, 种群规模为 50。多个方法的试验结果列于表(1)。分别执行各算法 100 次, 取其平均值作为实验结果。实验结果表明, 本文提出的混沌粒子群算法是有效的。

算法	遗传算法	粒子群算法	混沌粒子群算法
平均最小适应值	26828	8.81257	4.85437

表(1) 100 次运行的平均最小适应值比较

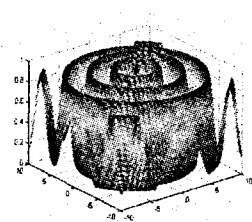
(二) Schaffer 函数

由于 Schaffer 函数为 GA-难度非常大的问题^[6], 因而经常被用来比较不同算法之间的性能。Schaffer 函数如下:

$$\max f(x) = 0.5 - \frac{(\sin \sqrt{x_1^2 + x_2^2})^2}{(1 + 0.001(x_1^2 + x_2^2))^2}, \quad x_i \in [-100, 100] \quad (8)$$

Schaffer 函数具有一个全局极大点 $x^*=(0,0)$, 其函数值 $f(x^*)=$

1。在全局极大点的周围, 无限多的局部极大点 (其函数值约为 0.9903) 形成一圈隆起将全局极大点包围, 如图(1)所示。由于 Schaffer 函数的强烈振荡性质以及它的全局极大点被无限多的局部极大点所包围的特性使得 Schaffer 函数的优化非常困难^[13]。



图(1) Schaffer 函数 ($x_i \in [-10, 10]$)

对于 Schaffer 函数, 搜索算法找到的最优解大于 0.9999 时, 认为寻优成功。因此, 可通过考察 100 次随机实验中搜索成功的次数来反映算法的优劣。

为便于比较, 最大进化代数分别取 200 和 500 两种情况, 从表(2)的实验结果可以看出, 混沌粒子群算法搜索成功的次数高于其他优化算法。因此该算法与其他算法相比, 具有更高效的搜索性能。

代数	群体规模	遗传算法	传统粒子群算法	混沌粒子群算法
200	15	0	9	13
	30	0	22	25
500	15	1	20	24
	30	3	38	43

图(1) Schaffer 函数 ()

实验环境: WinXP 平台, Visual C++ 6.0。

5. 加站问题最优优化算法研究

无线网络中, 基站的定位, 不仅决定了无线网络的服务质量, 而且极大地影响了连接/交换设备的数量, 从而决定了网络的建设质量。为解决网络规划方案中存在的覆盖盲区问题, 需要增加新基站。选择新基站站址和参数配置, 不仅要确保网络满足覆盖和容量等建设目标, 还不能引入其他网络问题。因此, 加站方案的设计需要综合考虑各种参数的相互关系, 在满足网络设计目标的同时将成本降至最小。为此, 定义覆盖率、覆盖重叠率两个指标来反映加站方案的质量。

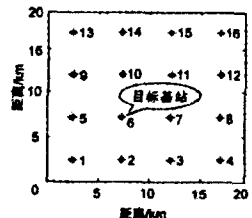
在本文的仿真计算中, 简化认为每个基站的覆盖范围为一正方形蜂窝, 覆盖范围的大小与基站的发射功率和移动台的最小接收功率有关, 路径损耗的计算是采用的 Hata 模型。定义覆盖率为被覆盖区域占规划区域的比例。覆盖重叠率定义为覆盖重叠的面积占规划区域的比例。规划方案应当满足覆盖率尽可能高同时覆盖重叠率尽可能低。为此, 定义目标函数^[14]为:

$$\text{Obj} = \beta_1(1 - \text{CovRate}) + \beta_2 \text{OvlRate}$$

其中, CovRate 为覆盖率; OvlRate 是覆盖重叠率; β_1, β_2 分别为这两个指标的权重, 而且满足: $\beta_1 + \beta_2 = 1$ 。目标值的高低在一定程度上反映了网络参数设计方案的综合质量, 目标值越低则方案越优。

混沌粒子群算法中, 粒子表示基站的坐标位置和发射功率等参数。

模拟仿真如下^[14]: 如图 2 所示, 其中标号为 6 的基站在原来规划方案中是没有的, 本文目的就是要通过搜索算法不仅可以找到 6 号基站的坐标位置, 而且能够找到 6 号基站的发射功率最佳配置, 达到最佳覆盖。实验中, 取粒子数为 50, $c_1=c_2=2.0$, $w_{\max}=0.9$, $w_{\min}=0.1$, $\beta_1=0.95$, $\beta_2=0.05$ 。



图(2) 目标基站站分布

多次模拟实验均表明, 本文算法收敛速度很快, 而且能够找到新加基站的地理位置和发射功率参数的全局最优解。

参考文献:

1. Fukuyama Y. Fundamentals of particle swarm techniques [A]. Lee K Y, El-Sharkawi M A. Modern Heuristic Optimization Techniques With Applications to Power Systems [M]. IEEE Power Engineering Society, 2002. 45-51.
2. 杨维. 粒子群优化算法综述[J]. 中国工程科学. 2004, 6(5): 87-94.

(下转第 83 页)

环;然后,将目标ip地址和当前路由器负责的地址范围进行比较。如果不属于那个范围,而且当前路由器是一个接入路由器时,那么可以肯定数据包是流向外网的,立刻跳出循环。如果某次遍历成功,那么就要按照链路的属性给数据包计算附加信息。

5.蜜罐系统构建

当蜜罐采用随机服务器部署方法时,是通过将多个蜜罐作为真实服务器的镜像安插在其周围,并提供相似的网络服务,使攻击者在随机扫描时难以辨别真实服务器和蜜罐,从而消耗其大量时间来寻找攻击目标。

该技术最一般的实现方法是在真实服务器周围部署多个物理蜜罐,但考虑到成本和管理方面,往往采用虚拟蜜罐进行部署,利用虚拟机在一个物理主机(宿主主机)上创建多个虚拟客户主机,并在每个客户主机上分别部署一个蜜罐。攻击者可以与虚拟主机的整个操作系统进行交互,从而记录更多的攻击行为。但是它的缺点在于受到物理主机内存容量和CPU能力的限制,只能创建有限个虚拟主机和虚拟蜜罐;一般不能接管一个C类网段中所有IP地址;而且作为高交互的蜜罐,不容易进行数据控制,给网络带来了高危险性。可以将此方案和采用此算法的蜜罐系统应用结合起来:在一个物理主机上创建多个客户主机,在其中的一些客户主机上安装采用此算法的蜜罐系统以接管大部分IP地址,并提供虚拟网络服务,而在剩下的客户主机上提供虚拟主机。这种方法将高交互和低交互的蜜罐结合在一起,分工和目的明确,各行其职。

结束语:

主动网络是新型的全新一代网络,目前基于主动网络的

研究还比较少,基于主动网络的蜜罐研究还属于待探索的全新领域^[4],本课题主要研究了基于主动网络环境 LINUX 系统下基于 IP 监听的蜜罐路由欺骗算法的蜜罐路由欺骗算法,在实现基于 IP 监听的蜜罐路由欺骗算法时改进了现有蜜罐系统,为 Honeyd 添加路由模块实现虚拟网络,改造 Honeyd 使之适应主动网络,并修改添加路由模块,增加蜜罐的欺骗性。同时在完善了 Honeyd 蜜罐系统的同时,还综合各种路由技术,运用 JAVA 技术编写了一个监控平台,组建了一个全新思想的完善的蜜罐网络,让这些蜜罐软件可以在主动网络上运行。课题提出的算法可以运用于主动网络安全领域,将有利于蜜罐技术的发展,以及在下一代新型网络中,蜜罐越来越广泛的应用于其网络安全体系,并以此来推动主动网络的发展。

参考文献:

- 1.熊华,郭世泽.网络安全——取证与蜜罐[M].人民邮电出版社,2003
- 2.马建峰,檀玉恒.蜜罐系统在入侵检测系统中的研究与设计[J].技术论坛,2003
- 3.张孟洋.蜜罐技术在入侵检测系统中的应用设计[J].丽水学院学报,2006
- 4.HoneynetProject.KnowYourEnemy:Honeynets [Z].http://project.honeynet.org
- 5.Lance Spitzner.The Value of Honeypots [Z]. http://www.securityfocus.com,2001
- 6.Jane Hillston, Leila Kloul. Performance Investigation of an On-Line Auction System. University of Edinburgh Kings Buildings Edinburgh [M], Scotland

(上接第73页)

(图二)左上方为原图像,右上方为噪声图像;坐下方为基于中值的反扩散方法一,右下方为本文改进的新方法二。方法一的 PSNR 为 25.91db,迭代 30 次;方法二的 PSSNR 为 31.07db,迭代 2 次。

3.总结与讨论:

在这篇文章中,我们对基于中值滤波的反扩散方法进行了改进,在每次扩散结束后又用中值滤波去除残留的椒盐噪声,这样不但没有增加运行时间反而更快和有效的实现了去噪的目的。本文建议的方法对一些低信噪比的分片光滑图像比较有效,如何利用改进的反扩散方法对其它类型的图像更加有效的进行混合噪声的处理还有待于进一步探讨。

(上接第77页)

- 3.Shi Y, Eberhart R C. Fuzzy Adaptive Particle Swarm Optimization[C]. In: Proceedings of the Congress on Evolutionary Computation, Seoul, Korea, 2001.
- 4.Ratnaweera A, Halgamuge S. Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients. IEEE Trans. Evolutionary Computation, 2004, 8 (3): 240-255.
- 5.王岁花.一类新颖的粒子群优化算法[J].计算机工程与应用. 2003 (13):109-111.
- 6.李敏强,寇纪淦,林丹,李书全.遗传算法的基本理论与应用[M].北京:科学出版社,2002.
- 7.杨俊杰.基于混沌搜索的粒子群优化算法[J].计算机工程与应用. 2005 (16):69-71.
- 8.Lovbjerg M, Rasmussen T K, Krink T. Hybrid Particle Swarm Optimiser With Breeding and Subpopulations [C]. In: Proc of the third Genetic and Evolutionary Computation Conference, 2001.
- 9.Ciuprina G, Ioan D, Munteanu I. Use of Intelligent-Particle Swarm Optimization in Electromagnetics [J]. IEEE Trans on Magnetics, 2002, 38

参考文献:

- 1.Omer Demirkaya, "Smoothing Impulsive Noise Using Nonlinear Diffusion Filtering", Springer Verlag Berlin Heidelberg, pp.111-122, 2004.
- 2.林宇辰,石青云, "一个能去噪和保持真实感的各项异性扩散方程", 计算机学报, vol 22, pp.1133-1137, Nov. 11, 1999.
- 3.P. Peoma and J. Malik, "Scale-space and edge detection using anisotropic diffusion," IEEE Trans. Pattern Anal. Machine Intell., vol. 12, pp. 629-639, July 1990.
- 4.J. Canny, "A Computational Approach to Edge Detection," IEEE Trans Pattern Anal. Machine Intell., vol. PAMI-8, pp. 679-698, 1986.
- (2): 1037-1040.
10. Van den Bergh F, Engelbrecht A P. Training Product Unit Networks Using Cooperative Particle Swarm Optimizers[C]. In: Proc of the third Genetic and Evolutionary Computation Conference, 2001.
- 11.李祥飞.基于混沌变量的前向神经网络结构优化设计[J].控制与决策. 2003, 18(6): 703-707.
- 12.刘自发.基于混沌粒子群优化方法的电力系统无功最优潮流[J].电力系统自动化. 2005, 29(7): 53-58.
- 13.李敏强,寇纪淦,林丹,李书全.遗传算法的基本理论与应用[M].北京:科学出版社,2002.
- 14.孙鹏,杨大成.无线网络规划的最优化算法研究[J].网络技术. 2005, 35(3): 14-16.
15. Eberhart R, Shi Yuhui. Particle Swarm Optimization :Development, Applications and Resources. IEEE International Conference on Evolutionary Computation, Seoul, 2001.